Parallel performance and benchmarking of the CE-QUAL-ICM family of three-dimensional water quality models

Mark R. Noel[1], Terry K. Gerald[2], and Carl F. Cerco[3]
[1]*AScI Corporation, U.S.A.*
[2]*AScI Corporation, U.S.A.*
[3]*U.S. Army Engineer Research and Development Center, U.S.A.*

Abstract

Accurate analyses of water quality issues pertaining to waterways require the application of eutrophication and contaminant transport/fate models to evaluate management alternatives. The movement of models to scalable, parallel computing platforms is a necessity since these simulations exhaust the resources of single processor computing systems.

The CE-QUAL-ICM family of three-dimensional water quality models, developed at the U.S. Army Engineer Research and Development Center Waterways Experiment Station (WES), Vicksburg, MS, consists of an eutrophication model (ICM) and a transport/fate model (ICM/TOXI). Both codes were parallelized by combining a single program multiple data (SPMD) execution model with data domain decomposition using the message passing interface (MPI) library. Two different domain decomposition strategies were tested for performance, a Hilbert Space-Filling technique from the Center for Subsurface Modeling, University of Texas at Austin and the METIS multi-level graph partitioning package from the University of Minnesota. Evaluating the parallel versions included obtaining performance statistics on three platforms: IBM-SP, Cray T3E, and SGI Origin 2000.

Results from the code parallelization effort indicate an order of magnitude decrease in model run-time can be achieved with as little as 16 processors. Furthermore, the application of these parallel codes to grids of varying resolution for the same test site indicate better performance can be obtained as the grid resolution increases.

Introduction

CE-QUAL-ICM (ICM), a water quality model developed at the U.S. Army Engineer Research and Development Center, Vicksburg, MS, models the transport and reaction of 20 or more state variables simultaneously [1]. Transport is based on flow and diffusion supplied by a three-dimensional hydrodynamic model, CH3D [2]. Hydrodynamics are computed once, stored on disk, and used repeatedly by ICM.

ICM has been utilized for long-term (most recently ten year simulations) eutrophication studies of Chesapeake Bay. These simulations would require hundreds of hours on Silicon Graphics workstations. Additionally, ten-year spin-up simulations were preferred to allow ICM to settle to a steady state. The requirements to complete these simulations would have been time prohibitive, thus, eliminating the scope and effectiveness of the study. Efforts by the Center for Subsurface Modeling (CSM) at the University of Texas at Austin allowed ICM to be parallelized and transported to massively parallel computing platforms [3]. Domain decompositions using both the Hilbert Space Filling Curve (HSFC) [4,5] and METIS strategies have been implemented to split the global domain into smaller sub-domains. Each sub-domain is processed by a single processor element (PE) and has local input and output files applicable only to that PE. Message passing tables and the MPI message passing interface are required for inter-processor communication. A pre/post processor (WQMPP) was developed for localizing input files and globalizing output files.

CE-QUAL-ICM/TOXI is a three-dimensional, finite volume, trace chemical/contaminant transport/fate model for surface water. ICM and ICM/TOXI share a common transport scheme. The difference between the two models is in the number and type of state variables simulated, and in the kinetic processes controlling trace chemicals as modeled in ICM/TOXI versus the eutrophication processes included in the ICM model.

The domain decomposition technique used for ICM/TOXI was adapted from the ICM parallelization project performed by the CSM at the University of Texas [6]. Similarly, one can use either the METIS or the HSFC method for sub-grid generation for making parallel runs with ICM/TOXI.

ICM/TOXI has been applied in a study of PCB fate/transport in the Hudson River located in the state of New York. Due to questions concerning the effect of grid density on predicted maximum concentration values, five grids of varying resolutions were created and applied to ICM/TOXI. Each grid was tested with a varying number of processors and execution times recorded for each computing platform.

The benefits of code parallelization can be summarized in two categories: simulation time reduction and higher grid densities. The ICM application to Chesapeake Bay requires a large number of scenarios to be run to determine the overall effect of changes in parametric variables. Therefore, run-time reductions are the main concern. The ICM/TOXI application is concerned more with the effect of grid resolution on simulated parameter values.

Hardware description

Parallel computing platforms used in this analysis are presented in Table 1.

Table 1. Parallel Computing Platform Specifications.

	Cray T3E	IBM SP 2	SGI Origin 2000
Number of Processors	520	255	112
Processor Speed	600 MHz	135 MHz	195 MHz
Computational Capacity	624 GFlops	137.7 GFlops	49.9 GFlops
Global Memory	256 Mbytes/ Cpu	256 GBytes	64 GBytes

Benchmarking of application ICM to Chesapeake Bay

The numerical model for Chesapeake Bay consists of 10,196 grid cells including 2,100 surface cells. The global domains, when decomposed by the HSFC and METIS strategies, produce vastly different sets of sub-domains. Utilizing these two strategies and varying the number of processors provided benchmark results across the three platforms.

After it was concluded identical results could be produced with serial or multiple processors across platforms, a 30-day test simulation was examined for benchmark results. The initial serial test was conducted on an SGI Challenge-M and used 5000 seconds of CPU time. On the massively parallel platforms, successive tests were performed doubling the number of processors from 2 to 32. Three statistics were examined for each test including wall clock time, number of times “speed-up”, and processor element efficiency.

The SGI Origin 2000 outperformed the Cray T3E, which outperformed the IBM SP regardless of the number of processors. The single processor test on the Origin used 3500 seconds of wall clock time; the T3E, 4900 seconds; the SP, 7900 seconds. With two processors, the time dropped to 1700 seconds for the Origin, 2800 for the T3E and 4600 for the SP. On all three test platforms, as the number of processors increased, the wall clock time for completion decreased. On the final test, when using 32 processors, all wall clock times dropped below 1000 seconds. The Origin used 167 seconds, while the T3E used 396 seconds and the SP used 779 seconds.

These three platforms were next compared by “speed-up.” Speed-up is the number of times faster a code runs in parallel rather than in serial. Speed-up is calculated by dividing the multiprocessor time into the single processor time.

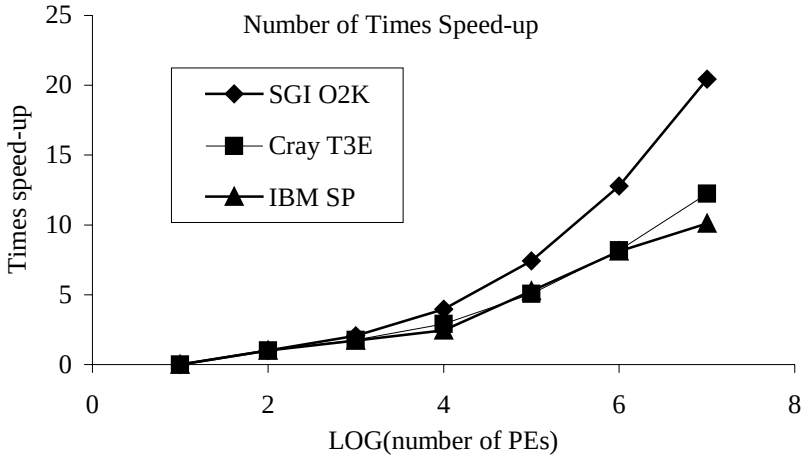

Figure 1: Number of Times Speed-up

Using the simulation times to illustrate the speed-up of the SGI Origin 2000, one processor used 3500 seconds while 32 processors used 160 seconds. The 20 times speed-up was derived by dividing 3500 by 160. Using 32 processors, the T3E achieved a 12 times speed-up and the SP realized a 10 times speed-up (Figure 1).

The last test used to compare these three platforms was processor element (PE) efficiency. PE efficiency, a number between 0 and 1, is determined by the ratio of single processor wall clock time to the number of processors multiplied by multiprocessor time. If a single processor run takes 700 seconds, then a 10 processor run with 1.0 PE efficiency will take 70 seconds. When using 32 processors, the Origin, with a PE efficiency of 0.64, was shown more adept than either the T3E or the SP, which both scored between 0.3 and 0.4. Interestingly, when using 2 processors, the Origin scored above optimum efficiency. After investigation, it was determined running in parallel can utilize the hardware of this massively parallel machine better than running with a single processor.

Benchmarking of application of ICM/TOXI to the Hudson River

Five grids of varying spatial resolution were used to determine what effect grid resolution has on predicted maximum PCB concentration calculations in a section of the Hudson River. This study also allows for an examination of how the parallel version of ICM/TOXI scales with varying grid resolution. Grid density was varied in the longitudinal (run of the river) dimension while the

lateral (across) dimension remained constant. Note all grids cover the exact same topography. As grid density is increased, the number of cells in the longitudinal dimension is increased thereby making each cell smaller in length. The grids are specified in Table 2.

Table 2. Grid specifications for the Hudson River.

Grid Name	Number of Cells	Longitudinal Size	Lateral Size	Number of Layers
20x20x5	1900	19	20	5
40x20x5	3900	39	20	5
80x20x5	7900	79	20	5
160x20x5	15900	159	20	5
320x20x5	31900	319	20	5

Results obtained from applying ICM/TOXI with METIS decomposition to each of the Hudson River test grids on the SGI Origin system are presented in Tables 3 and 4. Table 3 contains the actual run times in seconds obtained while Table 4 presents the speedup ratio obtained for each run relative to the respective serial run. Tables 5 and 6 present the same information for the Cray T3E system. Both the SGI Origin and the Cray T3E are accessed through a batch execution environment. Since each job must compete for resources such as processors, the maximum number of processors tested was 64. Attempting to allocate more processors results in very long queue wait times.

Table 3. ICM/TOXI on SGI Origin Timing Summary

Grid	1 Cpu	2 Cpus	4 Cpus	8 Cpus	16 Cpus	32 Cpus	64 Cpus
20x20x5	836	488	286	184	136	101	-----
40x20x5	1782	951	598	334	210	170	-----
80x20x5	4220	2099	1021	609	369	215	-----
160x20x5	8272	4141	2121	1139	676	422	-----
320x20x5	18285	8626	4217	2299	1074	633	405

Table 4. ICM/TOXI on SGI Origin Speedup Summary

Grid	2 Cpus	4 Cpus	8 Cpus	16 Cpus	32 Cpus	64 Cpus
20x20x5	1.71	2.92	4.54	6.15	8.28	----
40x20x5	1.87	2.98	5.33	8.49	10.48	----
80x20x5	2.01	4.13	6.93	11.43	19.63	----
160x20x5	2.00	3.90	7.26	12.24	19.60	----
320x20x5	2.12	4.34	7.95	17.03	28.89	45.15

Table 5. ICM/TOXI on Cray T3E Timing Summary.

Grid	1 Cpu	2 Cpus	4 Cpus	8 Cpus	16 Cpus	32 Cpus	64 Cpus
20x20x5	1318	747	432	276	197	136	----
40x20x5	3126	1468	951	577	256	195	----
80x20x5	5432	2864	1578	894	517	319	----
160x20x5	11151	5891	3003	1747	1015	530	----
320x20x5	*	11450	5913	3134	1597	986	1007

* Unable to execute ICM/TOXI for the 320x20x5 grid due to memory requirements.

Table 6. ICM/TOXI on Cray T3E Speedup Summary

Grid	2 Cpus	4 Cpus	8 Cpus	16 Cpus	32 Cpus
20x20x5	1.76	3.05	4.78	6.69	9.69
40x20x5	2.13	3.29	5.42	12.21	16.03
80x20x5	1.90	3.44	6.08	10.51	17.03
160x20x5	1.90	3.71	6.38	11.00	21.04

A study of Tables 4 and 6 leads to the observation that generally we can expect performance from parallel computing platforms to approach the ideal speedup limit as the computational grid density increases.

Note, the 320x20x5 test grid on the Cray T3E was unable to be run due to the memory requirements of the executable. However, this grid was run as a multiprocessor job. The Cray T3E does not support virtual memory because of the performance degradation it can cause. The maximum amount of memory available to the user is the physically installed amount per processor node less the operating system requirements. The SGI Origin platform is a virtual memory system. The user can select to disable paging when running interactively or use a queuing system that deliberately assigns a single task to a single processor for the entire duration of the run.

Hilbert Space Filling Curve versus METIS for ICM

The SGI Origin 2000 outperformed the Cray T3E and the IBM SP in benchmark tests. However other factors, such as the actual turnaround time, compiler options, queue wait time, efficient use of available hardware, or number of users accessing the machine, need to be considered when choosing a particular platform for in-depth water quality studies. Due to many of these factors, the Cray T3E, using only 32 processors, was the best choice for the Chesapeake Bay water quality study. Nevertheless, ICM will be applied to numerous other studies. Another testing of the parallelization process was needed to determine a maximum for the number of processors that could be used.

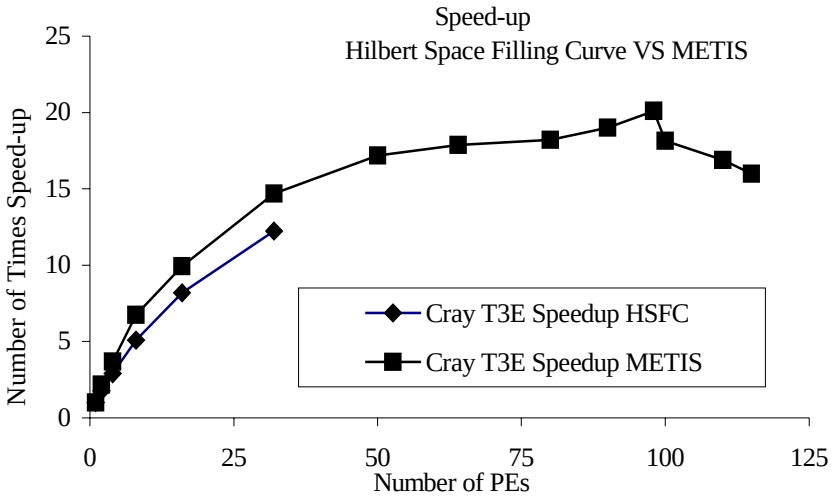

Figure 2: Speed-up HSFC versus METIS

The Hilbert Space Filling Curve (HSFC) decomposition strategy provided sub-domains with vastly different sizes, which caused load-balancing errors. As the number of processors increased, some local domains were of zero size limiting the number of processors that could be used with the HSFC to 39. The METIS strategy was implemented and provided a much better distribution of load balance to all processors. With an even load balance, the number of processors that could be utilized increased. Thus, processing times were lowered.

Another test, similar to the tests that compared different platforms, was performed on the Cray T3E comparing the HSFC and METIS decompositions. The HSFC strategy with 32 processors used 396 seconds to complete the 30-day simulation. The METIS strategy with 32 processors performed better, only using 330 seconds. In fact, when using METIS, processing times continued to decrease as the number of processors increased until more than 98 processors were used. The processor time at the maximum of 98 processors was 241 seconds (Figure 2).

When using more than 98 processors, times began to increase. Using too many processors cause each local PE to perform more message passing with other PEs when compared to the amount of time actually spent processing the local sub-domain.

Conclusion

Despite any difficulties or downtime in development and testing, the parallelization of CE-QUAL-ICM has resulted in significant increases in the simulation capability. In particular, studies of 10-year simulations that would have taken hundreds of hours on a serial machine are completed in less than a day. In fact, this parallel model is now in use for the newest nutrient reduction scenarios for the Chesapeake Bay Tributaries Study. The current depth and magnitude of this study simply would not be possible without a parallel code and platform. The CPU time constraints on a single CPU machine would limit the scope and volume of study available to scientists. The environmental and industrial impacts determined from this study require the most in-depth and conclusive information available.

The Chesapeake Bay nutrient reduction scenarios actually involve a full 20-year simulation. First, a 10-year spin-up runs allowing ICM to arrive at a steady state. The output from the spin-up is the initial conditions of the actual 10-year nutrient reduction scenario. This further emphasizes the need for practical simulation times (Table 7).

Table 7. Actual elapsed times (seconds) for simulations (20 years).

	10 year Spin-up	10 year nutrient reduction scenario
WQMPP (PREP)	1200	14,000 (3.9 hours)
Parallel CE-QUAL-ICM	35,000 (9.7 hours)	39,500 (10.97 hours)
WQMPP (POST)	86	2,550

While the theoretical speed-ups seem only moderate, the practical application to multiyear simulations and the information produced are invaluable. Compare the 140 hours or 5.8 days (280 hours for the spin-up and scenario) needed to simulate 10 years (using the serial code on the Silicon Graphics Challenge – M) to the 1 day needed using the parallel code. Including the spin-up, the serial code would take 2 weeks of valuable time to simulate one scenario. Using the parallel code, a complete spin-up and scenario can be simulated, verified and processed for World Wide Web examination by scientists across the country in less than one week.

For ICM/TOXI, the benefits from the parallelization effort have been clearly demonstrated. With a parallel version of the code available, users are given the capability of modeling much larger computational grids than have been previously possible.

Acknowledgments

Development of the Chesapeake Bay model package was sponsored by the U.S. Army Engineer District, Baltimore, MD and the Chesapeake Bay Program Office, U.S. Environmental Protection Agency. Additionally, this work was supported by the Department of Defense Common High Performance Computing Software Support Initiative.

References

[1] Cerco, C.F. and Cole,T.M. "Three-Dimensional Eutrophication Model of Chesapeake Bay", *Journal of Environmental Engineering*, Vol. 119, No. 6, November/December, 1993.

[2] Johnson, B., Kim, K., Heath, R., Hsieh, B., and Butler, L. "Validation of a three-dimensional hydrodynamic model of Chesapeake Bay.", *Journal of Hydrological Engineering*, ASCE, 199(1), pp. 2-20, 1993.

[3] Chippada, S., Dawson, C.N., Martinez, M.L., and Wheeler, M.F., "Parallel Computing for Finite Element Models of Surface Water Flow", *Computational Methods in Water Resources XI, Vol.2: Computational Methods in Surface Flow and Transport Problems*, eds. A.A. Aldama, J. Aparicio, C.A. Brebbia, W.G. Gray, I. Herrera & G.F. Pinder, Computational Mechanics Publications, Boston, pp. 63-70, 1996.

[4] Edwards, H.C. "A Parallel Infrastructure for Scalable Adaptive Finite Element Methods and its Application to Least Squares C Collocation", Ph.D. Thesis, The University of Texas, Austin, TX, May 1997.

[5] Sagan, H. *Space-Filling Curves*, Springer-Verlag, 1994.

[6] Chippada, S., Dawson, C.N., Parr, V.J., Wheeler, M.F., Cerco, C.F., Bunch, B.W., and Noel, M.R. "PCE-QUAL-ICM: A Parallel Water Quality Model Based on CE-QUAL-ICM", *Proceedings of the 1998 Department of Defense High Performance Computing Modernization Program Users Group Conference*, Houston, TX, Publications Branch, DoD HPC Modernization Office Technical Information Division, Washington D.C., June 1998.